UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/537,720	06/07/2005	Daniel Giambalvo	304393.02	5131

22971 7590 03/17/2011

MICROSOFT CORPORATION
ONE MICROSOFT WAY
REDMOND, WA 98052-6399

EXAMINER	
NELSON, CHRIS A	
ART UNIT	PAPER NUMBER
2193	

NOTIFICATION DATE	DELIVERY MODE
03/17/2011	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

ntovar@microsoft.com
p5docket@microsoft.com

Office Action Summary	Application No.	Applicant(s)	
	10/537,720	GIAMBALVO ET AL.	
	Examiner	Art Unit	
	CHRIS NELSON	2193	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *20 December 2010*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-16 and 18-20* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-16 and 18-20* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Response to Arguments

1. Applicant's arguments with respect to double patenting against claim 1 have been considered but are moot in view of the new ground(s) of rejection. Application 10/799351 has been patented, and a non-provisional double patenting rejection has been issued.

2. Applicant's arguments with respect to claim 1 have been considered but are moot in view of the new ground(s) of rejection in view of US 2004/0019889 A1.

Double Patenting

3. The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. A nonstatutory obviousness-type double patenting rejection is appropriate where the conflicting claims are not identical, but at least one examined application claim is not patentably distinct from the reference claim(s) because the examined application claim is either anticipated by, or would have been obvious over, the reference claim(s). See, e.g., *In re Berg*, 140 F.3d 1428, 46 USPQ2d 1226 (Fed. Cir. 1998); *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422

F.2d 438, 164 USPQ 619 (CCPA 1970); and *In re Thorington*, 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) or 1.321(d) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent either is shown to be commonly owned with this application, or claims an invention made as a result of activities undertaken within the scope of a joint research agreement.

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

4. **Claim 1** is rejected on the ground of nonstatutory obviousness-type double patenting as being unpatentable over claim 1 of U.S. Patent No. 7,853,609 in view of publication US 2004/0019889 A1. They are not patentably distinct from each other because the patented application includes similar functionality to the current application, as shown below.

Patent 7,853,609 discloses groups that may be allowed to receive updates at different times based upon which group they are in, as shown below. However, the patent does not claim a specific general group that receives updates only after another testing group has tested the update and an administrator has authorized it for general use. However, the examiner maintains that it was well known in the art at the time of the invention to do so, as taught by US 2004/0019889 (Melchione1).

Melchione1 discloses distributing software in stages, which are delivered to predefined groups of computers (Paragraphs 11, 57-61, and 111).

Therefore, it would have been obvious to implement at least two stages of software installation into the invention of the cited patent. The purpose for doing so would have been to reduce unanticipated problems caused by a new software release while still being able to take advantage of it's features (see Melchione1, paragraphs 6-8).

Current application	US 7,853,609
an update service node having an application programming interface for administering the distribution of software updates on the update service node, the application programming interface comprising:	A software update distribution system for distributing a software update over a communication network for distribution to client computers, comprising: a root update service node; and a plurality of child update service nodes operable to distribute software updates to client computers, wherein each of the plurality of child update service nodes comprises:
an update store for storing software updates	an update store for storing software updates
an update web service through which the update service node obtains software	an update web service through which the child update service node obtains

updates from a parent update service node over a communication network, and through which the update service node distributes software updates to child update service nodes over the communication network.	software updates from its parent update service node over the communication network, and through which the child update service node distributes software updates to its child update service nodes over the communication network;
an administration application programming interface (API) through which an administrator defines distribution groups including an evaluation group and a general group, and establishes distribution rules associated with each group, the distribution rules specifying the distribution of software updates to child update service nodes and client computers included in the respective distribution groups, the rules associated with the evaluation group specifying immediate distribution to the evaluation group, the rules associated with the general group specifying withholding distribution until authorization is received based on the evaluation, wherein the	wherein the root update service node includes a first administration application programming interface (API) and first administration user interface, wherein the first administration API and first administration user interface are operable to receive from an administrator a first set of rules for distributing software updates to at least some of the plurality of child update service nodes;

administration API is an object exposing a plurality of interface calls through which the administrator establishes said rules.	

Claim Rejections - 35 USC § 103

5. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

6. **Claims 1, 19, and 20** are rejected under 35 U.S.C. 103(a) as being unpatentable over Davis (US 6,282,712 B1) in view of East (US 2003/0061323 A1), in further view of Melchione1 (US 2004/0019889).

7. As per **claim 1**, Davis discloses an update service node having an application programming interface (Column 3, lines 38-50) for administering the distribution of software updates on the update service node (Column 4, line 52 through column 5, line 6, primary site), the application programming interface comprising:

a. an update store for storing software updates (Column 6, lines 31-35)

b. an update web service through which the update service node obtains software updates from a parent update service node (Column 4, line 52 through column 5, line 6, central site) over a communication network, and through which the update service node distributes software updates to child update service nodes (secondary site) over the communication network. More specifically, the central, primary, and secondary sites each contain a site server to provide update functionality (See column 5, lines 7-45).

Davis does not explicitly disclose a site server obtaining updates from another, beyond a brief mention that "the site server 202 stores software that can be installed on

other computers in the distributed system (Column 5, lines 16-18). However, the examiner maintains that it was well known in the art at the time of the invention to allow file servers to copy their update stores as shown by East.

In a similar field of endeavor, East discloses obtaining software updates from a parent update service node over a communications network, and distributing the software updates to child update service nodes over the communications network (See paragraph 0008, master & remote administrative servers in a control hierarchy)

The purpose for doing so would have been to reduce the time needed to perform updates (East 0008). This would be especially useful in combination with Davis because updates appear to only be available to site servers through the use of compact discs (Column 6, lines 52-56).

c. Davis further discloses an administration application programming interface (API) through which an administrator defines distribution groups including an evaluation group and a general group, and establishes distribution rules associated with each group, the distribution rules specifying the distribution of software updates to child update service nodes and client computers included in the respective distribution groups, the rules associated with the evaluation group specifying immediate distribution to the evaluation group, the rules associated with the general group specifying withholding distribution until authorization is received based on the evaluation, wherein the administration API is an object exposing a plurality of interface calls (Column 5, lines 7-32,

administrator's console) through which the administrator establishes said rules (Column 13, lines 43-44, administrator's preferences).

Davis and East disclose a framework for making groups and establishing distribution privileges to each group. However, Davis & East do not explicitly disclose an evaluation group and a general group, wherein the general group is only distributed updates after authorization has been received. However, the examiner maintains that it was well known in the art at the time of the invention to do so, as taught by Melchione1.

Melchione1 discloses distributing software in stages, which are delivered to predefined groups of computers (Paragraphs 11, 57-61, and 111).

Therefore, it would have been obvious to implement at least two stages of software installation into the invention of the Davis & East. The purpose for doing so would have been to reduce unanticipated problems caused by a new software release while still being able to take advantage of its features (see Melchione1, paragraphs 6-8).

8. **Claim 19** recites substantially similar limitations to claim 1, and is therefore rejected using the same art and rationale set forth above.

9. As per **Claim 20**, Davis further discloses requesting a product update catalog listing software updates available for distribution and selecting one or more software updates from the product update catalog (Column 13, lines 55-61).

10. **Claims 2-3 and 16** are rejected under 35 U.S.C. 103(a) as being unpatentable over Davis, East, & Melchione1 in view of Islam (US 7,219,964 B1).

11. As per **claim 2**, Davis, East, & Melchione1 disclose the update service node of Claim 1. Davis & East do not explicitly disclose wherein the configuration interface exposes a get configuration interface call which returns a configuration interface object for reading and writing software update administration configuration values to the update service node. However, the examiner maintains that it was well known in the art at the time of the invention to do so, as taught by Islam.

Islam discloses a set of configuration APIs using configuration mbeans (Column 9, line 35 through column 10, line 3).

It would have been obvious to use object based APIs to configure the site server, for the purpose of allowing a user to make changes in the configuration file using a GUI instead of a text editor.

12. As per **claim 3**, Davis, East, & Melchione1 disclose the update service node of Claim 2. Islam further discloses wherein the configuration interface object is an IConfiguration object (Column 9, line 35 through column 10, line 3). Islam's mbean is an obvious variant of an IConfiguration object.

13. As per **claim 16**, Davis, East, & Melchione1 disclose the update service node of Claim 1. Davis & East fail to disclose wherein the administration API is an IUpdateServer interface object. However, the examiner maintains that it was well known in the art at the time of the invention to do so, as taught by Islam.

Islam discloses a set of configuration APIs using configuration mbeans (Column 9, line 35 through column 10, line 3) which are used to update a server's configuration. This would have been an obvious variant of an IUpdateserver interface object.

It would have been obvious to use object based APIs to configure the site server, for the purpose of allowing a user to make changes in the configuration file using a GUI instead of a text editor.

14. **Claims 4-9 and 11-15** are rejected under 35 U.S.C. 103(a) as being unpatentable over Davis, East, Melchione1 & Islam in view of Melchione2 (US 2003/0200300 A1).

15. As per **claim 4**, Davis, East, Melchione1 & Islam disclose the update service node of Claim 2. The above cited references do not explicitly disclose subscription or subscriptions APIs. However, the examiner maintains that it was well known in the art at the time of the invention to do so, as taught by Melchione2.

Melchione2 discloses subscribing to a set of updates (paragraph 156). Information based on the subscription is made available using a configuration interface (paragraph 16-18, and 140-141). In combination with the above cited references, this information could be delivered as an mbean object. The purpose for doing so would have been to allow users to enter into contracts and automatically have their software updated.

16. As per **claim 5**, Davis, East, Melchione1, Islam & Melchione2 disclose the update service node of Claim 4. Melchione2 further discloses the update service node of Claim 4, wherein the subscription interface object is an ISubscription interface object (paragraph 16-18, and 140-141). Islam's mbean in combination with Melchione2's subscription interface is an obvious variant of an ISubscription interface object.

17. As per **claim 6**, Davis, East, Melchione1, Islam & Melchione2 disclose the update service node of Claim 4. Melchione2 further discloses the update service node of Claim 4, wherein the administration API exposes a get subscriptions interface call which returns a subscription collection interface object defined on the update service node (paragraph 16-18, and 140-141).

18. As per **claims 7-8**, Davis, East, Melchione1, Islam & Melchione2 disclose the update service node of Claim 4. Davis further discloses wherein the administration API exposes a get update interface call which returns a update interface object corresponding to an update identifier passed in the get update interface call (Column 13, lines 55-58). In combination with Islam's mbeans, this would be an obvious variant of an IUpdate interface call.

19. As per **claim 9**, Davis, East, Melchione1, Islam & Melchione2 disclose the update service node of Claim 7. Davis further discloses wherein the administration API exposes a get updates interface call which returns an update collection object containing update interface objects corresponding to values passed in the get updates interface call (Column 13, lines 55-58).

20. As per **claims 11-12**, Davis, East, Melchione1, Islam & Melchione2 disclose the update service node of Claim 9. Islam further discloses wherein the administration API exposes a get computer interface call which returns an client computer object corresponding to the a client computer associated with the update service node and that was identified in the get computer interface call (Islam, column 12, lines 35-62, and more specifically "server configuration may contain information for standalone server

instance 920F"). This information would be accessible using the configuration API of column 9, lines 20-67. This would also be considered an obvious variant of an IComputer interface call using mbeans.

21. As per **claim 13**, Davis, East, Melchione1, Islam & Melchione2 disclose the update service node of Claim 11. Islam further discloses wherein the administration API exposes a get computers interface call which returns a computer collection object including client computer objects corresponding to client computers associated with the update service node (Islam, column 12, lines 35-62, and more specifically "while configuration 1000B may be associated with servers 920E-F"). This information would be accessible using the configuration API of column 9, lines 20-67.

22. As per **claims 14**, Davis, East, Melchione1, Islam & Melchione2 disclose the update service node of Claim 13. Islam further discloses wherein the administration API exposes a get group interface call which returns an target group object that was identified in the get group interface call (Islam, column 12, lines 35-62, and more specifically "while configuration 1000B may be associated with servers 920E-F"). This information would be accessible using the configuration API of column 9, lines 20-67.

23. As per **claims 15**, Davis, East, Melchione1, Islam & Melchione2 disclose the update service node of Claim 14. Islam further discloses wherein the administration API exposes a get groups interface call which returns a target group collection object including target group objects corresponding to target groups on the update service node (Islam, column 12, lines 35-62, and more specifically "Likewise, cluster configuration 1010A may contain information for all servers 920A-E executing in cluster

900"). This information would be accessible using the configuration API of column 9, lines 20-67.

Claim 10 is rejected under 35 U.S.C. 103(a) as being unpatentable over Davis, East, Melchione1, Islam & Melchione2 in view of Sierer (US 2004/0255291 A1).

As per **claims 10**, Davis, East, Melchione1, Islam & Melchione2 disclose the update service node of Claim 9. Islam allows updates to be hidden based on usability (natural language, operating system, etc), but does not explicitly have a boolean value in the interface call. However, the examiner maintains that it was well known in the art at the time of the invention to do so, as taught by Sierer.

Sierer further discloses wherein the values passed to the get updates interface call include a deployed state object and an exclude hidden updates Boolean value (Paragraph 237). Specifically, Sierer allows deployed objects to be hidden based on display information. The purpose for doing so would have been to allow a user to only see applicable updates that have not been installed yet.

24. **Claim 18** are rejected under 35 U.S.C. 103(a) as being unpatentable over Islam, Melchione2, Davis, & Melchione1.

As per **claim 18**, Islam discloses a software update distribution system for distributing software updates, the software update distribution system comprising:

an update service node (figure 6, Administration client 600, and associated text)

and an administration application programming interface (API) associated with the update service node, wherein the administration API is an interface object exposing a plurality of interface calls for controlling the distribution of software updates (Column 9, lines 20-67, the administration API including:

a create computer target group through which at least two target groups (Column 12, lines 27-63) are defined including an all-computers group and an evaluation target group for evaluating software updates prior to distribution to the all-computers group;

Islam does not explicitly disclose an all-computers group and an evaluation target group. However, the examiner maintains that it was well known in the art at the time of the invention to do so, as taught by Melchione1.

Melchione1 discloses distributing software in stages, which are delivered to predefined groups of computers (Paragraphs 11, 57-61, and 111). Therefore, it would have been obvious to implement at least two stages of software installation into the invention of the Davis & East. The purpose for doing so would have been to reduce unanticipated problems caused by a new software release while still being able to take advantage of its features (see Melchione1, paragraphs 6-8).

a get configuration interface call which returns a configuration interface object for reading and writing software update administration configuration values to the update service node (Column 9, line 35 through column 10, line 3);

a get subscription interface call which returns a subscription interface object defined on the update service node;

Melchione2 discloses subscribing to a set of updates (paragraph 156). Information based on the subscription is made available using a configuration interface (paragraph 16-18, and 140-141). In combination with Islam, this information could be delivered as an mbean object. The purpose for doing so would have been to allow users to enter into contracts and automatically have their software updated.

a get update interface call which returns an update interface object corresponding to an update identifier passed in the get update interface call;

Davis further discloses wherein the administration API exposes a get update interface call which returns a update interface object corresponding to an update identifier passed in the get update interface call, and a get updates interface call which returns an update collection object containing update interface objects corresponding to values passed in the get updates interface call (Column 13, lines 55-58). The purpose for doing so would have been to allow an administrator access to a catalogue of updates to aid in administration.

a get updates interface call which returns an update collection object containing update interface objects corresponding to values passed in the get updates interface call;

Davis further discloses wherein the administration API exposes a get update interface call which returns a update interface object corresponding to an update identifier passed in the get update interface call, and a get updates interface call which returns an update collection object containing update interface objects corresponding to values passed in the get updates interface call (Column 13, lines 55-58). The purpose for doing so would have been to allow an administrator access to a catalogue of updates to aid in administration.

a get computer interface call which returns a client computer object corresponding to the a client computer associated with the update service node and that was identified in the get computer interface call (Islam, column 12, lines 35-62, and more specifically “server configuration may contain information for standalone server instance 920F”). This information would be accessible using the configuration API of column 9, lines 20-67.

a get computers interface call which returns a computer collection object including client computer objects corresponding to client computers associated with the update service node (Islam, column 12, lines 35-62, and more

specifically "while configuration 1000B may be associated with servers 920E-F"). This information would be accessible using the configuration API of column 9, lines 20-67.

a get group interface call which returns a target group object that was identified in the get group interface call (Islam, column 12, lines 35-62, and more specifically "while configuration 1000B may be associated with servers 920E-F"). This information would be accessible using the configuration API of column 9, lines 20-67.

a get groups interface call which returns a target group collection object including target group objects corresponding to target groups on the update service node (Islam, column 12, lines 35-62, and more specifically "Likewise, cluster configuration 1010A may contain information for all servers 920A-E executing in cluster 900"). This information would be accessible using the configuration API of column 9, lines 20-67.

Conclusion

Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within

TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to CHRIS NELSON whose telephone number is (571)270-7256. The examiner can normally be reached on Monday to Thursday, 9AM to 5PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Lewis Bullock can be reached on (571)272-3759. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic

Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/CHRIS NELSON/
Examiner, Art Unit 2193

/Lewis A. Bullock, Jr./
Supervisory Patent Examiner, Art Unit 2193